

CURSO:	ENGENHARIA DE SOFTWARE		
DISCIPLINA:	Estruturas de Dados e Algoritmos	TURMA	A
SEMESTRE:	2013.2	CRÉDITOS:	4
PROFESSOR:	Giovanni Almeida Santos		

LISTA DE EXERCÍCIOS 4

ÁRVORES BINÁRIAS

Para a resolução dos exercícios abaixo, você terá que implementar uma estrutura para representar uma árvore. Além disso, as seguintes funções deverão ser construídas:

inserir() – recebe dois parâmetros, um ponteiro para a árvore e o elemento a ser inserido
 pre_ordem() – recebe um ponteiro para a raiz da árvore e, recursivamente, mostra os elementos em pré-ordem
 em_ordem() – recebe um ponteiro para a raiz da árvore e, recursivamente, mostra os elementos em ordem
 pos_ordem() – recebe um ponteiro para a raiz da árvore e, recursivamente, mostra os elementos em pós-ordem
 profundidade() – recebe um ponteiro para a raiz da árvore e retorna a profundidade (altura) da árvore, recursivamente
 contar_folhas() – recebe um ponteiro para a raiz da árvore e retorna o número de folhas da árvore, recursivamente
 mostrar_folhas() – recebe um ponteiro para a raiz da árvore e mostra as folhas da árvore, recursivamente

23) A profundidade (altura) de uma árvore é representada pelo valor do maior nível existente na árvore. Dessa forma, faça um programa que leia um número inteiro N ($N \geq 0$) e uma sequência de números inteiros, armazenando-os em uma árvore binária de busca. A saída do programa deverá ser o valor da profundidade (altura) da árvore seguido de um caractere de nova linha ('\n'). Considere que o primeiro nível da árvore (nível da raiz) é o valor 1.

Entrada		Saída
5	35 12 20 78 44	3
1	35	1
0		0
12	7 14 45 67 23 8 19 3 29 34 90 10	6

24) Em árvore binária, uma folha é representada por um nó que não possui filhos. Com base nisso, faça um programa que receba um número inteiro N ($N \geq 0$) e uma sequência de números inteiros, armazenando-os em uma árvore binária de busca. A saída do programa deverá ser o número de folhas contidas na árvore, seguido de um caractere de nova linha ('\n'). A tabela abaixo mostra exemplos de entradas e saídas para o programa.

Entrada		Saída
5	35 12 20 78 44	2
1	35	1
0		0
12	7 14 45 67 23 8 19 3 29 34 90 10	5

25) Em árvore binária, uma folha é representada por um nó que não possui filhos. Com base nisso, faça um programa que receba um número inteiro N ($N \geq 0$) e uma sequência de números inteiros, armazenando-os em uma árvore binária de busca. A saída do programa deverá ser a lista dos valores contidos nas folhas presentes na árvore, separados por espaço em branco, seguida de um caractere de nova linha ('\n'). Por questões de simplificação, deverá haver um espaço em branco após o último número da lista. A tabela abaixo mostra exemplos de entradas e saídas para o programa.

Entrada		Saída
5	35 12 20 78 44	20 44
1	35	35
2	90 90	90
12	7 14 45 67 23 8 19 3 29 34 90 10	3 10 19 34 90

26) Em árvore binária, uma folha é representada por um nó que não possui filhos. Com base nisso, faça um programa que receba um número inteiro N ($N \geq 0$) e uma sequência de números inteiros, armazenando-os em uma árvore binária de busca. O programa deverá mostrar a lista de valores contidos na árvore utilizando a travessia de pré-ordem.

Entrada		Saída
5	35 12 20 78 44	35 12 20 78 44
1	35	35
2	90 90	90
12	7 14 45 67 23 8 19 3 29 34 90 10	7 3 14 8 10 45 23 19 29 34 67 90

27) Em árvore binária, uma folha é representada por um nó que não possui filhos. Com base nisso, faça um programa que receba um número inteiro N ($N \geq 0$) e uma sequência de números inteiros, armazenando-os em uma árvore binária de busca. O programa deverá mostrar a lista de valores contidos na árvore utilizando a travessia em ordem.

Entrada		Saída
5	35 12 20 78 44	12 20 35 44 78
1	35	35
2	90 90	90
12	7 14 45 67 23 8 19 3 29 34 90 10	3 7 8 10 14 19 23 29 34 45 67 90

28) Em árvore binária, uma folha é representada por um nó que não possui filhos. Com base nisso, faça um programa que receba um número inteiro N ($N \geq 0$) e uma sequência de números inteiros, armazenando-os em uma árvore binária de busca. O programa deverá mostrar a lista de valores contidos na árvore utilizando a travessia de pós-ordem.

Entrada		Saída
5	35 12 20 78 44	20 12 44 78 35
1	35	35
2	90 90	90
12	7 14 45 67 23 8 19 3 29 34 90 10	3 10 8 19 34 29 23 90 67 45 14 7